Project Deliverable Report

Deliverable D4.3.1 – Learner positioning service version 1.5

Work Package: WP4
Task: 4.1
Date of delivery: Contractual: 01-09-2010 Actual: 15-10-2010
Code name: D4.3.1 Version: 1.0 Draft ☐ Final ☑
Type of deliverable: Report
Sec.(distribution level): Public
Contributors: Gaston Burek, Robert Koblischke, Dale Gerdemann, Christoph Mauerhofer, Kiril Simov and Petya Osenova
Authors (Partner): Gaston Burek, Robert Koblischke, Dale Gerdemann, Christoph Mauerhofer, Kiril Simov and Petya Osenova
Contact Person: Gaston Burek
WP/Task responsible: Gaston Burek
EC Project Officer: Ms. M. Csap

Abstract (for dissemination): The goal of task 4.1 is to position the learner with respect to a curriculum. The actual positioning is done on the basis of analysis of the learner's answers to a set of questions. The services within task 4.1 are integrated with the services within the task 6.1 in order to provide access to learning materials appropriate for reading by the learners after the positioning and to tutor in the process of the creation of the questions for the positioning task.

Keywords List: Learner positioning, learner network, language use, ontology

LTfLL Project Coordination at: Open University of the Netherlands
Valkenburgerweg 177, 6419 AT Heerlen, The Netherlands
Tel: +31 45 5762901 – Fax: +31 45 5762800
Table of Contents

EXECUTIVE SUMMARY.................................................................3

1 INTRODUCTION AND METHODOLOGY BACKGROUND..........................5

2 POSITIONING SERVICE VERSION 1.5 FEATURES.................................7

2.1 STEP BY STEP DESCRIPTION OF THE POSITIONING SOLUTION AS IMPLEMENTED IN VERSION 1.5 8
  2.1.1 Tutor user responsible for setting up and maintaining the service .......... 8
  2.1.2 Learner user................................................................. 12
  2.1.3 Tutor user responsible for positioning .................................. 14

3 FINAL SOFTWARE DOCUMENTATION: WIDGETS, SERVICES, AND DATA ........16

3.1 PEDAGOGIC ORIENTATION ......................................................16
3.2 SYSTEM OVERVIEW: ARCHITECTURE........................................17
3.3 CHANGES IN VERSION 1.5 ....................................................... 19
3.4 HOW DOES VERSION 1.5 ADDRESS VALIDATION RESULTS AND REVIEWERS COMMENTS? ... 22

4 TRANSFERABILITY ........................................................................24

4.1 PEDAGOGICAL, ORGANISATIONAL AND TECHNICAL .......................24

5 EVALUATION: BETA TESTING VERIFICATION FOR VERSION 1.5 ..........25

5.1 DEFINITION OF VERIFICATION ..................................................25
5.2 METHODOLOGY AND RESULTS...................................................26

6 CONCLUSION................................................................................27

REFERENCES..................................................................................29

APPENDIX 1: OVERVIEW DOCUMENTATION ON SERVICE APIs PROVIDED AND DATA FORMATS.........................................................30
Executive Summary

In this deliverable we describe the implementation of version 1.5 of the LTfLL positioning service. We explain how the new version of the service addresses first round validation results and reviewers’ criticism over the previous version and additionally we provide details about ongoing verification plans. We describe the service architecture, report improvements over version 1.0 and discuss transferability.

Finally, we discuss ‘additional’ positioning service functionalities and usability features that are outside the scope of this project but can be easily be implemented and incorporated into in the current version as part of a thread with other LTfLL services.

Description of the integration between the knowledge poor and knowledge rich approaches

The LTfLL positioning service implements two approaches i.e. knowledge poor and knowledge rich. While the knowledge poor approach supports the positioning of the learner by means of analysing the learner’s language usage (quantitative and qualitative), the knowledge rich approach supports the positioning of the learner by means of investigating the conceptual coverage of the learner’s texts. Version 1.5 incorporates to the positioning service functionalities from the Formal Learning Support System (FLSS – task 6.1) to support the (semi-)automatic integration of both approaches – knowledge poor approach provides suggestions for new lexicalization of concepts. These new lexicalizations will support in better way the semantic annotation. Task 6.1 service functionalities allow the positioning service to automatically annotate learner answers and learning materials with ontology concepts and allows tutors to refine that annotations. New lexical knowledge derived from that process is formalised (semi-)automatically in the ontology lexicon.

Description of plan for verification

The ongoing verification of version 1.5 of the positioning service includes testing the three live feedback outputs. The ‘Answer Score’ that is the percentage value for the quality of the learner answer, the ‘Distinct Phrases’ that provide information about the the learners use of language and finally the lists of Common, Missing and Additional concepts supporting the evaluation of the learner conceptual coverage. Verification results will be reported in LTfLL deliverable D7.4.

Final version

Version 1.5 is the final version of the positioning service as a standalone application. A new and fully widgetised version of the service as part of a thread with task 6.1 services will be released together with the submission of LTfLL deliverable 2.4. This
'threaded' version will include additional functionalities (e.g. semantic search for learning materials) in addition to version 1.5 functionalities.
1 Introduction and methodology background

Our implementation plan for version 1.5 of the positioning service has been informed by validation results and reviewer comments about version 1.0 (see section 4.3). The plan included the further development of the service as a separate service and the incorporation of functionalities from task 6.1 service (e.g. manual annotation interface). Below we provide a short description of version 1.5 and outline the developments along the lines of our implementation plan.

The knowledge poor approach scores phrases extracted from text belonging to a Community of Practise (CoP) according to distinctive features of their usage (Burek and Gerdemann, 2009). Those ‘utterances’ are then used to build a semantic space for the Latent Semantic Analysis (LSA; Landauer, 1998) of learner texts. More specifically, the service measures the cosine distance between learner answers and stored answers that have been already graded by tutors. The outputs of this analysis include a list of relevant phrases written by the learner and a list of missing phrases. Additionally, it includes a grade on a chosen scale for each learner answer submitted for analysis.

The knowledge rich approach assumes a curriculum with respect to which the learner is positioned. The curriculum defines a set of topics which the learners have to know. The goal of knowledge rich approach is to determine the level of conceptual coverage between the requirements of the curriculum and the learners’ knowledge. In the current settings, the curriculum topics are represented as a questionnaire. The questionnaire is a set of questions which determine the necessary aspects of the learners’ knowledge. The learners have to demonstrate their knowledge by answering the questions. To give a basis for the positioning, the tutor, who prepares the questionnaire on the basis of the curriculum, explicates the expected knowledge as a set of concepts assigned to each question. The knowledge rich approach positions learners by automatically identifying concepts within the learner answers and comparing them to the concepts assigned to the questions. Evidence of the conceptual coverage of the learner text is displayed by the service in the form of concept lists where each concept is represented by its lexicalisation.

The integration of FLSS functionalities within the positioning service is aimed at providing access to semantically annotated learning materials in order to support the creation of questionnaires and the search of appropriate learning materials.

The integration involves data communication between both services. This communication facilitates the semi-automatic lexicalisation of the ontology used by the knowledge rich approach. Users of the positioning service (i.e. tutors) can inspect the learning materials in search for knowledge poor output (phrases) not available in the ontology lexicon. Using FLSS, tutors can then manually annotate those phrases with concepts from the ontology. The manual annotation of the leaning materials can be done after the positioning service has completed the automatic annotation.

Those manual annotations supported by FLSS are temporally stored and are accessible to the service every time the same user is logged in. The stored annotations are used by the positioning service knowledge rich approach for string matching on top of the automatic analysis of learner’s answer. Periodically a service
administrator will incorporate in the ontology lexicon the stored annotations as concept lexicalisations. Configuration of the positioning service is done by using web-interfaces and the associated text management service that allows the uploading of text in various format extensions (e.g., .txt, .doc, .pdf) or entering URLs as indicated in visual interface.
2 Positioning service version 1.5 features

Version 1.5 similarly to the previous version of the services is based on the notion of learning network described in (Kalz et al. 2007). A learning network incorporates different actors and resources – learning institutions, tutors, learners, learning materials, other resources. In such a learning network the new (to some subject) learner has to be positioned with respect to a target competence defined by a curriculum, set of learning materials, etc. The services of task 4.1 explicate learner’s competence and compare it to the target competence. Here under competence we understand the following definition:

Characteristics that individuals have and use in appropriate, consistent ways in order to achieve desired performance. These characteristics include knowledge, skills, aspects of self-image, social motives, traits, thought patterns, mind-sets, and ways of thinking, feeling and acting. (Dubois et al 2004)

Of course the 4.1 services can not cover the whole range of elements in this definition of competence. The services cover the learner’s skills to use domain specific language (Knowledge Poor Approach) and the conceptual knowledge of the learner (Knowledge Rich Approach). This is done by analyzing the learner answers to questions written by tutors. Questions, learner answers and learner materials are parts of the learning network that is accessible to learners during the positioning. The conceptual knowledge within the learning network is represented as an ontology and it is explicated in the other components of the learning network via concept annotation. The language usage is represented within appropriate learning materials. All these sources are used to support the positioning services of task 4.1.

The positioning service provides live ‘on the fly’ feedback on learner answers by means of two approaches i.e. knowledge poor and knowledge rich approaches already described in section 1.

After studying the output of the positioning service for a learner answer, tutors can examine the learning materials annotated with relevant ontology concepts and decide which of those materials need to be studied by the learner, and additionally in which area of conceptual knowledge the learner may require further support. As in version 1.0 learners can use the output of the positioning service to evaluate their own position and identify their strengths and weaknesses.

The positioning service can automatically annotate learning materials with concepts from the ontology. Tutors can visually inspect those annotations and after studying the knowledge poor output (phrases list) for the relevant question they can edit the annotations in search for those phrases. Tutors can edit annotations by requesting the FLSS annotation studio. Ontology lexicalisations originated in tutor manual annotations are temporally stored for matching phrases within learner answers. This additional analysis of learners answers identifies conceptual coverage that has not been already identified by the knowledge rich approach. Additionally, the question tagging functionality to be implemented for the threaded and widgetised future version (2.0) of the positioning service will allow tutors to manually tag questions with
ontology concepts. The tutor will be supported in this task by selecting the concepts from the list of concepts identified in learning material, consequently reminding the tutor to provide learning material for all the required concepts.

2.1  Step by step description of the positioning solution as implemented in version 1.5

This section provides a step by step description of the positioning solution from the user perspective.

2.1.1  Tutor user responsible for setting up and maintaining the service

- The tutor create a course (see figure 1) and write questions (see figure 2) that cover relevant ontology concepts.
**D.4.3.1 Learner Positioning Service version 1.5**

![Learner Positioning Service](image)

**Figure 1 Course management screenshot**

**Figure 2 Question input box screenshot**
- For each question tutors can upload text or enter URLs of learning materials covering the relevant concepts (see figure 3).

![Edit Question (ID 78)](image)

**Figure 3  Relevant learning materials upload screenshot**

- The tutor can request annotation of the learning materials. Each text is automatically annotated by the service using the relevant ontology and associated lexicon (see figure 4).
Figure 4 Automatic learning material annotations screenshot
The tutor can inspect, edit and modify (add or delete) the automatic annotations using the annotation studio (see figure 5). Tutor manual annotations are stored temporarily in a database and can be accessed by the positioning service via provided interfaces queried using web-services. The service uses the ontology lexicon and tutor manual annotations to evaluate learner answer conceptual coverage (see figures 6 and 7).

Figure 5  Annotation studio screenshot

2.1.2 Learner user
- The learner answers a question and requests live feedback from the service. The service analyses the answer and returns outputs (see figure 6) from the knowledge poor qualitative analysis (positive and missing phrases list), a knowledge rich concept coverage analysis (common, missing and additional concepts list) and a LSA based text similarity analysis for the learner answer (answer score).
- The learner can edit his/her answer and request live feedback as many times as needed to improve it.
- The learner can access his/her grade and personalised tutor feedback once the tutor has completed the steps described in the subsection below.

2.1.3 Tutor user responsible for positioning
- For each learner answer the tutor requests live feedback and analyse the output (see figure 7).
- The tutor grade the answer and write personalised feedback taking into account the live feedback. Tutor can also refine (manually) the list of phrases (see figure 7) shown to the learner.
Figure 7 Learner live feedback and phrase refinement screenshot
3 Final software documentation: Widgets, Services, and Data

3.1 Pedagogic orientation
The functionality of the ‘Learner positioning service’ is addressing different pedagogic goals during the lifelong learning process.

The primary purpose of the service is to determine the current knowledge of the learner regarding the next courses she/he will attend. As described in the LTfLL 4.1 scenario the results of the positioning service are useful for the tutor to build the learning plan based on the existing/missing knowledge and available learning materials. Typically one tutor is responsible for the positioning of many learners and in this case the service enables the tutor to save time and avoids latency till he is able to serve the individual learners.

Learners present their existing knowledge in “her/his own words (writing sentences)” and avoids the use of existing answers (e.g. during multiple choice tests). This way of presenting information correlates with real life requirements to the learner in his existing or for future job.

The service ‘Live Feedback’ provides immediate feedback to learners’ answers based on the language used in the CoP (Community of Practice) and conceptual coverage. Using these results learners updates (improve) their answers – this procedure offers an additional learning process: The use of the language for the CoP for this domain is been trained.

The positioning service offers an additional pedagogic benefit: ‘Motivating the learner’.

As additional output the ‘Live Feedback’ presents the missing concepts and phrases in learners’ answers. This information illustrates gaps in learners’ knowledge existent at the moment and motivates him to move on finding appropriate learning materials (therefore also tools of the LTfLL project are useful).

This experience of ‘Self-Positioning’ and ‘Self-Learning’ guides the individuals into an improved lifelong learning process, where they are also willing to perform individual self-learning without interaction with tutors or other individuals.

All over, the ‘Learner positioning service’ provides different pedagogic benefits:

1. In addition to the improved positioning task the tool raises learner’s interest for new knowledge in the specific domain.
2. The required assistance by tutors is reduced or for some parts of the learning process eliminated (the autonomy of the learner is strengthened, costs of learning are reduced).
3. The learner gets more familiar with the language used in the CoP.
3.2 System overview: architecture

The system architecture can be divided into the course management system and its storage facilities and the various feedback methods utilising the data (e.g. learning materials) the tutor uses as input to the course management system.

Image 8 shows a use-case diagram, describing the basic functionalities of the positioning service. It contains both a tutor and a student view/user interface, which was explained in greater detail in deliverable 4.2. The data structure from version 1.0 remains largely unchanged, however, version 1.5 has some additional features such as the versioning of learner answers or the logging of feedback activities which can be used as additional input for the upcoming evaluation.

![Figure 8 Positioning service use-case diagram](image)

The quantitative feedback of the knowledge poor approach utilizes latent semantic analysis (LSA) in order to compute the similarities between a learner answer to other graded answers and derive an automated score/grade from there. To accomplish this, the service uses the WP2 space management system to automatically generate suitable spaces for the specific course and (if already available) learning materials and learner answers.
This system mainly builds on R\(^1\), Weka\(^2\) and SVDLIBC\(^3\); the “tm”\(^4\) R-package for text mining, corpus creation and pre-processing (interfacing some Weka functions) of the sparse document-term-matrix, the SVDLIBC library to calculate the sparse singular value decomposition (SVD) as well as the “lsa”\(^5\) R-package for the actual document comparisons based on the dimensionality reduced matrices. However, the “tm” and “lsa” packages had to be modified because in order to compute SVDs of large matrices the “lsa” package had to be able to handle the sparse matrices provided by “tm” and the “tm” package had to store and pass on additional data required for the consecutive LSA computations. This not only required the implementation of interfaces between the two packages, but also some additional adaptions to the “tm” package for increased efficiency and effectivity. The current implementations can be viewed and downloaded from the LTfLL sourceforge page\(^6\) and it is planned to merge them into the current version of the “tm” package to enable LSA computations natively for this package.

The score of a new learner answer is calculated by folding in the answer and the already existing answers into the generated space and comparing the resultant document vectors. Currently a document is graded with the (similarity-) weighted average of the grades of the five closest answers, with a diminished grade if these similarities fall below the threshold.

Each question can be trained for the semantic space, so the ideal number of space dimensions is computed to achieve the highest correlation possible between human grades and system grades. Additionally all the existing answers are already folded in the space and stored by the system so this computationally expensive step only has to be performed for the new learner answer to be analyzed and not for the existing answers its compared against. This strategy enables feedback response times of less than two seconds in our scenario even for bigger spaces (30k+ unique terms, 400 dimensions).

The knowledge poor qualitative feedback builds on phrase extraction based on maximal suffix arrays and computing their distinctiveness based on their frequencies in either highly or lowly graded answers. This is based on the Java phrase extraction “Saphre”\(^7\) (suffix arrays for phrase extraction) library which is being used to extract phrases from texts and was developed for this project. Those texts are automatically selected by the service depending on the course (and learning materials and student answers if already existing) and data (e.g. frequency vector representations) provided by WP2. Phrases are loaded into R to parse the graded learner answers.

---

\(^1\)The R Project: http://www.r-project.org/
\(^2\)Weka 3: http://www.cs.waikato.ac.nz/ml/weka/
\(^3\)SVDLIBC Library: http://tedlab.mit.edu/~dr/SVDLIBC/
\(^4\)Text Mining (tm) Package: http://tm.r-forge.r-project.org/
\(^5\)Latent Semantic Analysis (lsa) Package: http://cran.r-project.org/web/packages/lsa/index.html
\(^6\)tm and lsa Packages (modified for LTfLL): http://ltfll.svn.sourceforge.net/viewvc/ltfll/v2/wp2/R/
\(^7\)Suffix Arrays for Phrase Extraction (Saphre): http://saphre.sourceforge.net/
and calculate a score for each phrase and question based on the average grade of the answers containing the phrase, the residual inverse document frequency from the domain documents as well as the document frequencies in the answers. The tutor also has the choice to mark phrases as irrelevant if “Saphre” failed to do so. Then, the service knowledge poor approach output displays the 10 highest scoring phrases contained in the answer, as well as the 10 highest scoring phrases not contained in the answer. Additionally before presenting it to the user it filters these phrase lists to reduce redundancies to a minimum.

The knowledge rich feedback annotates with concepts the learners’ answers to the questions in the questionnaire. Then the annotation is compared with the assigned by the tutor concepts for each question. In order to being able to provide feedback about which question-relevant concepts are present/missing in an answer, for each question the relevant ontology concepts have to be known. We have extended the service to cover other languages like Bulgarian. For version 1.5 these relations were hard-coded into the system, but for version 2 it will be possible to assign them manually via the user interface. However, since the course management system also allows to assign learning material to each question, the concepts discovered in each learning material that are relevant to that particular question are suggested by default and may be edited. This also enables direct recommendations of which learning material the student still has to read (that contains missing concepts). The thread between 4.1 and 6.1 includes these automated annotation services, optional manual re-editing of annotations via the WP 6.1 annotator interface and a storage facility for suggestions of new lexicalisations and annotations by the tutor.

The user interface is implemented in HTML and Javascript generated by PHP scripts executed by an Apache server and makes heavy use of AJAX technologies to dynamically load different fragments of the interface. Nearly all components of the course management system can be browsed, while for modifications (i.e. new courses/questions/answers/learning materials can be created, edited and deleted) a series of XML web services can be referenced via HTTP requests. Also the program logic behind the different feedback systems is queried via similar web services implemented either in PHP or R (enabled by the RApache module). The necessary data is stored and queried in a MySQL database. The programming language used for the relevant 6.1 services was Java in combination with an Apache web server.

The eventual widgetisation and repackaging of the user interface into a Wookie widget once version 2 is finished will provide means of easier installation to any compatible environment. Structurally, it is planned that these widgets will be divided into one course management widget and the various feedback widgets. Until now, all user interface parts are widgets integrated in a stand-alone version, which should greatly facilitate the future decoupling of individual parts as widgets.

3.3 Changes in version 1.5

Rapache: http://biostat.mc.vanderbilt.edu/rapache/index.html
For version 1.5 a focus was set on improving the quality for all feedback versions, as well as deepening the integration with task 6.1 and illustrate that in a “short thread”. For the knowledge poor feedback we have implemented an evaluation framework to quantify the reliability of the output score and provide a toolkit to monitor changes in grading when tuning the different parameters. Initial results from using this toolkit have already yielded significant improvements of the human-to-machine score correlations (up to 0.6) by adding algorithms that find the ideal number of LSA dimensions on a per-question basis. However, as of to date no cross validation is performed yet as each grade is calculated on a N-1 training set.

For the qualitative feedback, a new version of our phrase extraction software “Saphre” (as published on Sourceforge) was deployed and interfaced on our server. This allowed us to separate the phrase extraction and phrase score generation steps, thus being able to use a larger sample of domain specific texts for phrase extraction, not necessarily needing to be graded. This reduced the setup costs of the qualitative feedback dramatically, as now more phrases are sampled with less or no graded material required. To score these phrases graded answers are still required, but they are now scored on a per-question basis, leading to more reliable results.

For the knowledge rich feedback, an interface has been implemented to increase the integration of the feedback which was previously based on hard-coded ontology concept-question relations and concept lexicalisations. Now, questions can be manually annotated by selecting the automatically extracted concepts (as interfaced with task 6.1) from the related learning material. For the feedback, the concepts contained in the answer are being extracted by a web service provided by task 6.1 and compared to those concepts related to or required by the question. This allows for alterations to the knowledge rich feedback by the tutor and to use it for different and new questions, integrating the positioning service with 6.1 task services in the process. Also, in case certain lexicalisations turn out to be missing, task 6.1 has provided a direct interface to an annotation editor, enabling the tutor to suggest new lexicalisation to the service.

The user interface has also be enhanced, such as improved usability and simplicity of the feedback training facilities for the tutor, enabling the possibility to assign learning material directly to questions instead of courses, improved cross-browser compatibility to name a few. The textual data resources (domain specific and generic documents for various languages) have been growing as well to allow for a potentially better “knowledge poor” feedback. Additional versioning and logging functionality has been added as well to gain further insight from the upcoming user evaluation process.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Version 1.5 - Done</th>
<th>Version 2 - Work in Progress</th>
<th>Goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge Poor (KP) - Quantitative Score</td>
<td>Add automated training feature that determines the ideal number of</td>
<td>Add technical evaluation framework to allow for an automated validation of</td>
<td>Improved feedback quality</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Objective</td>
</tr>
<tr>
<td>Knowledge Poor (KP) - Distinct Phrases</td>
<td>Knowledge Rich (KR) - Concepts covered</td>
<td>User Interface</td>
<td></td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>---------------------------------------</td>
<td>---------------</td>
<td></td>
</tr>
<tr>
<td>LSA dimensions for each question</td>
<td>Generate phrases from domain specific documents, not only graded documents</td>
<td>Simplify semantic space and phrase extraction training interface</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Generate phrase scores not in relation to whole course or CoP, but instead on a per-question basis</td>
<td>Improve facilities to upload new learning material from the web and common document formats</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Add additional phrase filtering before user presentation</td>
<td>Move learning material frame from course view to question view,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Slightly lower setup costs (less graded answers required)</td>
<td>Wookie/Elgg Widgetization</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Improved and more intuitive list of phrases</td>
<td>Lower setup costs through less NLP (LSA, Phrase extraction, etc.) expertise required for the tutor, easier upload of learning material</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Add interface for automated (and optional manual re-) ontology concept extraction of learning material and learner answers</td>
<td>Make way for learning material suggestions on a per-question basis (see KR improvements)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Add interface to allow manual assignment of required ontology concepts to a question instead of hard-coded concept/question relations</td>
<td>Internet Explorer compatibility</td>
<td></td>
</tr>
</tbody>
</table>
allow assigning relevance of learning material per question
Improve cross browser compatibility
Minor usability changes like more intuitive browsing of the answers lists in the tutor view, etc.

Data and Back-end changes
Increase document base, number and variety of domain specific and generic documents
Add versioning of answers
Add additional system logging and data collection to live feedback requests
Increased document base for potentially improved feedback (KP)
Keep previous versions of graded answers and use them for KP training as well
Allow for collection of additional validation relevant data regarding the usage of live feedback

3.4 How does version 1.5 address validation results and reviewers comments?

Some of the new functionalities described in the previous section are our response to first round validation results (Armitt et al., 2010) and reviewer criticisms to version 1.0 of the service.

In version 1.5 the output results have been improved by extending existing repository of ‘domain specific text’ to all service functionalities.

On the knowledge poor side phrase extraction has been improved by increasing the number of texts available for the phrase extraction and therefore increasing the size of the sample of suffix arrays available for calculating the relevant statistic, i.e. phrase scores. In version 1.0 the mentioned repository was used only in the LSA based answer score feedback. Additionally, phrase scores for version 1.5 are
calculated only for ‘positive phrases’. This improvement facilitates the interpretation of the output by allowing the user to focus on characteristic phrases rather than on ‘positive’\(^9\) and ‘negative’\(^{10}\) phrases as in the previous version.

On the knowledge rich side the ontology lexicon coverage is now improved by allowing the service to use tutor annotation as lexicalisations for concepts in the ontology.

In addition to language technology based improvements we have also enhanced learner’s interpretation competency for the results of the positioning service. Motivating the learner is one of the important benefits resulting from using the service. To achieve this goal the new version uses the wording ‘common’ and ‘missing’ for phrases and concepts. In addition, during training learners receive now a more detailed introduction on how to interpret service outputs.

Because of this better understanding of the service output it is easier for the learner to find appropriate learning materials with the help of other LTfLL tools or other search functionalities. As a side effect of these activities we expect that the ‘trust value’ of the service will increase during the next round of validation.

The overall effort for setting up the positioning service for a specific domain in a learning environment has been reduced because the installation, setup-time and the amount of required initial data have been reduced. Regarding these changes we are expecting that the service will be cost efficient for smaller groups (less than 100 learners depending on the specific learning environment).

\(^9\)Positive phrases were extracted from high quality text.
\(^{10}\)Negative phrases were extracted from low quality text.
4 Transferability

4.1 Pedagogical, organisational and technical

As described in LTFL D8.5 (Mauerhofer et al., 2010) education providers can deploy the positioning service in diverse education environments. Training provider institutions such as Bitmedia, secondary schools and e-learning providers can adapt the main core functionalities of 4.1 + 6.1 services to their own environment. Bitmedia, in particular, is planning to implement a commercial version of the positioning service in its own learning environment, and secondary schools in Austria are interested in testing a version of the service in their environment.

Bitmedia is also planning to test a version of the positioning service in an e-learning environment. Since the system was design to easily adapt to different learning environments this should not problem given the availability of sufficient learner texts to train the service.

The service will work best in environments where tutors understand that learning is a social process and therefore they emphasise the communicative aspects of their domain including domain specific terminology, proper phraseology, etc. It is highly desirable that tutors have a certain degree meta-linguistic awareness in the sense that they are aware of their own speech genre (Bakhtin, 1986). It should be emphasised, however, that tutors also by using the service will further develop awareness of such linguistics issues. Even in a highly technical domain communication skill are very important. We know that speech genres and communication skill are of utmost importance in mathematics and some of the best best books/articles on domain specific language use are for mathematics e.g. Maier and Schweiger (1999) and Wells (1999). Of course, if the formula to text ratio is very high the feedback from the positioning service will be of limited use.

Bitmedia’s commercial version of the service will require minor adaption. To implement the positioning service in an e-learning environment, it may be necessary re combine widgets depending to the concrete company policy. Available widgets can be displayed according to functionality related requirements.
5 Evaluation: Beta testing verification for version 1.5

Version 1.0 of the positioning service already has an appropriate user interface, which was easy to use for the learners and tutors (Armitt et al., 2010). Version 1.5 of the service provides improvements for the functionality and tuning capabilities for optimizing service outputs. We are currently completing the verification in English and Bulgarian for this latest version of the service.

5.1 Definition of verification

The positioning service offers three outputs to the learner and tutor, which are being individually measured during the ongoing verification.

The first output of the service is the ‘Answer Score’, which is a percentage value for the quality of the answer given by the learner. During the traditional positioning this value was assigned by the tutors.

For one specific answer there are different scores given by the individual tutors. But all these scores are in a specific range (e.g. between 60% and 70%). Therefore we do not expect that the ‘Answer Score’ of our service will exactly meet the score given by one individual tutor.

The output of the service should provide a value, which is within the range.

During verification of the service answer scoring functionality tutors evaluate learner answers and give a grade. The grade given is compared to the score given by the service.

The results for the ‘distinct phrases’ (knowledge poor output) provides information for ‘positive’ and ‘missing’ phrases in the learners answer. This output informs the learner about the knowledge she/he has covered in the answer and provides ‘ideas’ of additional phrases which are used by others for answering this question.

In a similar form the service lists the common, missing and additional concepts (knowledge rich output), where the missing concepts are used as hints for the learner to find learning materials.

The verification of these outputs (phrases and concepts) is done with two aims. First of all, the provided missing phrases or concepts have to be really relevant for answering this question and therefore have an impact on the answer grading. To verify the relevance of the phrases and concepts lists a knowledge expert in relevant domain is required. Therefore, the verification of the accuracy of the lists is being carried out by the tutors. Tutors are asked to improve a set of answers from the language usage and conceptual coverage perspective using knowledge poor and
knowledge rich outputs. We then compare the answer score given by the service to the original answer and the score given by the service to the improved answer.

As a second requirement, the information provided by the service output should be useful for the learner to find additional learning materials.

5.2 Methodology and Results

The positioning service has built on technologies that verify the accuracy of the output and tune the language technology based results of the service. During the use of the service the ‘Answer Score’ calculated by the system is used as first suggestion for this grading. The final grading for the learner is given by a tutor. In this way each time an answer is graded by a tutor, data is added to the system for the future calculations of the ‘Answer Score’.

In the user interface the ‘train lsa’ and ‘train phrases’ functionality has been added to update the semantic space.

The verification during the alpha testing of the software was done by comparing the originally calculated ‘Answer Score’ from the system with the answer score given by the tutor. Following the circumstance that also the individual grades given by the tutors are within a range we verified that the calculated result by the service is within +/- 15% of the tutors grading.

The new version of the positioning service improve accuracy for the list of phrases by allowing tutors to deselect (not accept) the phrases in the list. Additionally, the new version also improves the results of the concept in the ontology by providing suggestions for new concepts on the basis of the learning materials.

Tutors are able to edit the data used by the service. Because these updates on the ontology require knowledge in language technologies in addition to the domain specific knowledge we decided that the updates are not directly stored in the data platform for the 6.1 service. The tutors will create update requests for the background data which will be proofed by a language technology specialist. During the validation we will verify if the tutors are able to produce suggestive updates for the background data.

During the validation round the accuracy of the ‘live feedback’ will be measured based on internal logging and based on the feedback given by learners and tutors. Therefore, the service will log the provided ‘Answer Score’ by the system followed up by the final score given by the tutor. Using these values we are able to calculate the accuracy of the service output compared to the tutors rating.
To measure the accuracy of the list of distinct phrases the primary measures will be provided by the questionnaires used for the tutors. In addition to this feedback we will have a look at the number of not accepted phrases by the tutors. To verify the accuracy of the list of concepts also tutors’ feedback will be used to calculate the number of incorrect entries in the list compared with the number of all entries in the list.

6 Conclusion

In this deliverable we have presented the version 1.5 of the positioning service and explained its integration with 6.1 task services (in particular with the annotation studio) supports the integration between the knowledge poor and knowledge rich approaches to positioning.

We have described the new functionalities presented in this version of the service from two perspectives: usability and technical point of view. We have detailed our ongoing verification plan and explained that its result will be included in deliverable D7.4. In this deliverable we have also addressed the transferability of the service from the organisational and technical point of view.

Version 1.5 of the positioning service outlines the first round validation results and reviewers comments. In its initial conception, the positioning service was intended as an evaluative task. The service would determine the competence level of the learner, and then find some appropriate learning materials (in the ‘zone of proximal development’ for that learner). Fairly early in the project, it was recognised that this kind of positioning service by itself would be of a limited use both for tutors and learners. The central problem is that the quality of a fully automatic positioning cannot be guaranteed. Thus, it was decided that the positioning service should focus on supporting tutors in positioning the learner and in addition - on giving formative feedback to the learner.

Moreover, the positioning service supports learners in creating their own learning records, and tutors to build up course materials repositories as well as to lexicalise the ontology that formalises the relevant conceptual knowledge to be covered by learners. Data generated by the positioning service can be used to feed the rest of the LTFL services. In particular, complementary conceptual coverage derived from the knowledge rich analysis of learner answers can be used for grouping learners and building ‘collaborative teams of learners’, and therefore releasing the workload of tutors.

Once the ‘collaborative teams of learners’ are a conform group, they can be given an assignment to work on and to discuss in a forum. At least the following two advantages are envisaged:
1. Students are encouraged to think more deeply about their language use.
2. More text is generated, which can possibly be mined, and thus – give better orientation within the learning process.

The resulting forum entries can be fed to the LTfLL service PolyCAFe (deliverable 5.3) that maps topic threads in educational chat forums. Here, the idea is not to evaluate the forum participants, but rather to monitor the ongoing usefulness of the forum and thus ensure that the forum is promoting the learning as a social activity (see in Vygotsky (1978) and generally in social constructivist theory).

PolyCAFe can monitor the process of learning as a social activity within the forums. Similarly, learning texts and instructional materials can be used by the LTfLL service CONSPECT (deliverable 4.3.2) to evaluate learner emerging conceptual development. The LTfLL 6.2 task services could use the ontology lexicalisations to expand the lexicon further and facilitate the creation and terminology mapping between folksonomies. Additionally, 6.2 services can use the learning materials and learner generated texts to provide ongoing help to the learner by the use of social constructed folksonomies, which are used to mine for relevant learning materials.

In order to create greater quantities of text for the service, a couple of other assignments are given. Students are required to write individual texts on various specified topics. And as another simpler exercise, they are given a longer, finished text. Then they are required to make some editing to make it sound more like the way they would have expressed themselves.

As a result of the text mining carried out on the learners texts, a parallel phrase model is constructed and represented as parallel suffix arrays. Editings are classified as either grammatical, stylistic or content changes. The appropriate editings are incorporated into a personal, grammar/style checker.
References


Christoph Mauerhofer, Kamakshi Rajagopal, Alisdair Smithies, Martin Krippner, Gillian Armitt and all LTfLL partners (2010). LTfLL project deliverable 8.5.


Kalz, Marco; Van Bruggen, Jan; Rusman, Ellen; Giesbers, Bas; Koper, Rob. (2007). Positioning of Learners in Learning Networks with Content, Metadata and Ontologies. In Interactive Learning Environments, Volume 15, Issue 2 August 2007 , pages 191 – 200


Appendix 1: Overview documentation on service APIs provided and data formats

The following is a description of the feedback-relevant web services. There are a large number of web services that are not part of the main core of the positioning service implementation and will not be presented here, i.e. the course management system or the training of the different feedback services. All web services are to be invoked either via HTTP GET or POST requests and return XML data.

Below we describe the main five web services, the three live feedback related web services and the two web services facilitating automated annotation and the manual editing/suggesting of new lexicalisations implemented for the 4.1 - 6.1 thread. The complete file structure of the 4.1 code is available on Sourceforge¹¹.

<table>
<thead>
<tr>
<th>Name</th>
<th>KP - Quantitative Feedback</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description</strong></td>
<td>Retrieve grade estimate for submitted answer.</td>
</tr>
<tr>
<td><strong>URI</strong></td>
<td><a href="http://augur.wu.ac.at/v2/wp4.1/webservices/lsa.rws">http://augur.wu.ac.at/v2/wp4.1/webservices/lsa.rws</a></td>
</tr>
<tr>
<td><strong>Method</strong></td>
<td>REST / GET+POST</td>
</tr>
</tbody>
</table>
| **Parameter** | action : string "grade"  
method : string "weighted"  
question_id : integer [references questions]  
content : string |
| **Output Format** | XML |
| **Output Sample** | `<WSR:webServiceResponse>
<grade>
<method>cosine weighted mean grade</method>
<space_id>107</space_id>
<weighted_grade>0.80035778546166</weighted_grade>
</grade>
</WSR:webServiceResponse>` |

<table>
<thead>
<tr>
<th>Name</th>
<th>KP - Qualitative Feedback</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description</strong></td>
<td>Retrieve distinct phrases list for submitted answer.</td>
</tr>
<tr>
<td><strong>URI</strong></td>
<td><a href="http://augur.wu.ac.at/v2/wp4.1/webservices/positioning.rws">http://augur.wu.ac.at/v2/wp4.1/webservices/positioning.rws</a></td>
</tr>
<tr>
<td><strong>Method</strong></td>
<td>REST / GET+POST</td>
</tr>
</tbody>
</table>
| **Parameter** | action : string "list_scores"  
question_id : integer [references questions]  
content : string |

¹¹WP4.1 Sourceforge repository: http://ltfll.svn.sourceforge.net/viewvc/ltfll/v2/wp4.1/
Output Format | XML
--- | ---
**Output Sample** | `<?xml version="1.0" encoding="ISO-8859-1"?>
<WSR:webServiceResponse>
<phrases>
<max_score>19.5252355486432</max_score>
<content>
central process unit: consists of registers, arithmetic unit (alu), addressing unit.
a cpu can have multiple cores with varying clock rates and different sizes of cpu cache. the most popular consumer cpus are manufactured by intel and amd.
</content>
<phrase id="1">
<text>core</text>
<score>19.5252355486432</score>
<accept>1</accept>
<start>0</start>
</phrase>
<phrase id="2">
<text>register</text>
<score>17.236859162272</score>
<accept>1</accept>
<start>0</start>
</phrase>
...
</phrases>
</WSR:webServiceResponse>`

<table>
<thead>
<tr>
<th>Name</th>
<th><strong>KR - Ontology Concepts Covered</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description</strong></td>
<td>Retrieve list of required concepts covered by the answer.</td>
</tr>
<tr>
<td><strong>URI</strong></td>
<td><code>http://augur.wu-wien.ac.at/v2/wp4.1/webservices/knowledgerichproxy.php</code></td>
</tr>
<tr>
<td><strong>Method</strong></td>
<td>REST / GET+POST</td>
</tr>
</tbody>
</table>
| **Parameter** | **action** : string "getConcepts"  
**qid** : integer [references questions]  
**answer** : string |
| **Output Format** | XML |
| **Output Sample** | `<answer qid="48">  
<eval>5</eval>  
<ann>  
<common-concept>  
<concept uri="http://www.loa-cnr.it/ontologies/OWN/OWN.owl#UNIT_OF_MEASUREMENT__UNIT">unit</concept>  
<concept uri="http://www.lt4el.eu/CSnCS#AdvancedMicroDevices">AMD, Inc.</concept>  
...</common-concept>  
</answer>` |
<missing-concept>
<concept uri="http://www.loa-cnr.it/ontologies/OWN/OWN.owl#RATE_2">frequency</concept>
<concept uri="http://www.lt4el.eu/CSnCS#BIOS">BIOS</concept>
<concept uri="http://www.lt4el.eu/CSnCS#ChipSet">chipset</concept>
...
</missing-concept>
</answer>

<table>
<thead>
<tr>
<th>Name</th>
<th>Automated document annotator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Identify lexicalisations and annotate them with their respective concepts.</td>
</tr>
<tr>
<td>URI</td>
<td><a href="http://213.191.192.50:8080/wp4annotation/annotate">http://213.191.192.50:8080/wp4annotation/annotate</a></td>
</tr>
<tr>
<td>Method</td>
<td>POST</td>
</tr>
<tr>
<td>Parameter</td>
<td>REQUEST BODY: string [content to be annotated] HEADER PARAMETERS: lang: string [language iso code i.e. “en”] id: integer type: string [“answer” or “learning_material” as related to WP4.1]</td>
</tr>
<tr>
<td>Output Format</td>
<td>XML</td>
</tr>
<tr>
<td>Output Sample</td>
<td>&lt;root&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;par&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;s&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latfc&quot; id=&quot;t652&quot; sp=&quot;y&quot;&gt;Power&lt;/tok&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latas&quot; id=&quot;t653&quot; sp=&quot;y&quot;&gt;supply&lt;/tok&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;Concept class=&quot;http://www.loa-cnr.it/ontologies/OWN/OWN.owl#UNIT_OF_MEASUREMENT__UNIT&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latas&quot; id=&quot;t654&quot; sp=&quot;y&quot;&gt;units&lt;/tok&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;/Concept&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latas&quot; id=&quot;t655&quot; sp=&quot;y&quot;&gt;used&lt;/tok&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latas&quot; id=&quot;t656&quot; sp=&quot;y&quot;&gt;in&lt;/tok&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;Concept class=&quot;http://www.lt4el.eu/CSnCS#Computer&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latas&quot; id=&quot;t657&quot; sp=&quot;y&quot;&gt;computers&lt;/tok&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;/Concept&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latas&quot; id=&quot;t658&quot; sp=&quot;y&quot;&gt;are&lt;/tok&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latas&quot; id=&quot;t659&quot; sp=&quot;y&quot;&gt;nearly&lt;/tok&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latas&quot; id=&quot;t660&quot; sp=&quot;y&quot;&gt;always&lt;/tok&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;Concept class=&quot;http://www.lt4el.eu/CSnCS#Switch&quot;&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latas&quot; id=&quot;t661&quot; sp=&quot;y&quot;&gt;switch&lt;/tok&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;/Concept&gt;</td>
</tr>
<tr>
<td></td>
<td>&lt;tok class=&quot;latas&quot; id=&quot;t662&quot; sp=&quot;y&quot;&gt;mode&lt;/tok&gt;</td>
</tr>
</tbody>
</table>
Name | Lexicon suggestions service  
Description | List all lexicalisation suggestions submitted to the annotation editor.  
URI | http://213.191.192.50:8080/wp4annotation/lexmanager/suggest  
Method | GET  
Parameter | userID: string  
userMail: string [optional]  
conceptID: string [optional]  
lexEntry: string [optional]  
lang: string [language iso code i.e. “en”]  
Output Format | XML  
Output Sample | <suggestions>  
<entry status="new">  
<time>Fri Sep 17 19:07:42 EEST 2010</time>  
<userID>alex</userID>  
<conceptID>http://www.lt4el.eu/CSnCS#MemoryBoard</conceptID>  
<lexEntry>memory</lexEntry>  
<lang>en</lang>  
</entry>  
</suggestions>