A TEXTBOOK-BASED SERIOUS GAME FOR PRACTISING SPOKEN LANGUAGE

Claudia Baur[1], Manny Rayner[1], Nikos Tsourakis[1]

[1]*Université de Genève (SWITZERLAND)*

Abstract

We describe a web-enabled serious game intended to help German-speaking beginner students of English improve their generative and auditory competence. The tool is built on top of the existing CALL-SLT platform, the architecture of which is briefly described. It offers a short course of 8 interactive lessons using a combined vocabulary of about 450 words, and was developed in collaboration with a secondary school teacher, with the content taken from a commonly used textbook. Lessons are structured as a short dialogue between the student and the machine, where the student is encouraged to use simple language in practical contexts like booking a hotel room, buying clothes, or ordering a meal in a restaurant.

A lesson connects together groups of component examples. At each turn, the system starts by playing a short video file in English, and simultaneously displays a piece of text in German indicating to the student how they should reply. The students give a spoken response in English; if they are uncertain how to respond, they can first ask for a help example. After the student answers, the system performs speech recognition, machine translation and matching, and either accepts or rejects. There are multiple paths through the dialogue, controlled by a lesson structure defined in an XML file supplied by the course designer. The overall system is gamified to increase student motivation, using common game elements such as badges.

We conclude with a brief description of an informal pre-study carried out on several children aged between 9 and 13. The older children liked the system and found it easy to use, while the younger ones struggled.

Keywords: Computer Assisted Language Learning (CALL), Speech Recognition, Dialogue-Based Translation Game, Second Language Acquisition (SLA), English as a Second Language (ESL).

1 INTRODUCTION

In today’s world, Second Language Acquisition (SLA) is becoming increasingly important, especially for non-English natives. English is used as a lingua franca both in business and leisure time activities. It is essential for Swiss citizens to have some basic knowledge of English if they wish to communicate with foreign business partners or go on a holiday abroad. The most efficient method for learning a second language is – according to today’s understanding – direct communication with native speakers, favourably linked with an immersion into the country’s culture and habits. However, this approach is time-consuming and cost-intensive. This is why in recent years there has been an increased interest in Computer Assisted Language Learning (CALL) programmes, which allow the learners to practise their language skills at any time and place without paying a lot of money. Such CALL systems classically offer a variety of exercises. In this paper we present a CALL tool with integrated speech recognition, which allows the learners to practise their receptive and productive skills in English.

The CALL-SLT tool is based on the concept of Wang and Seneff’s Translation Game [1] and offers Germanophone beginner students of English a dialogue-based speech environment in which they can practise their language skills. The novelty in this project is that the various lessons are based on the textbook used by Swiss schools and can hence be directly integrated in the curriculum. Furthermore the system has been developed in close collaboration with Swiss English teachers, who gave their feedback and input throughout the development process.

From a pedagogical point of view, the system is supposed to support the pupils in their language learning, helping them to build up the self-confidence which is necessary to communicate freely with an English native. The CALL approach seems to be especially suitable to do this, due to the neutral environment a computer represents. In discussions with English teachers, various lessons were developed, which can help immerse students in real-world conversations.

1.1 Overview of architecture

CALL-SLT [2] is a prompt-response system, in which the system and the user take alternate turns. Each interchange begins with the system giving the student a prompt, which in the present version is a combination of an English-language multimedia file and a piece of German text; for example, at the beginning of the Hotel lesson, the system might play a cartoon clip of a desk clerk asking “How many nights would you like to stay at our hotel?”, simultaneously showing the German text “Frag: Zimmer für 6 Nächte” (“ask: room for 6 nights”). The student then gives a spoken response; the intent is that this should be reasonably free, so here one can for instance answer “I would like to stay for six nights”, “I want a room for six nights”, “A room for six nights please”, and several other variants.

The system decides whether to accept or reject the response by first performing speech recognition, then translating to a language-neutral (interlingual) representation, and finally matching this representation against the language-neutral representation of the prompt. A “help” button allows the student, at any time, to access a correct sentence in both written and spoken form. When the system has decided whether to accept or reject, it moves to a new state; the choice of state is determined by an XML-based script written by the course designer, which specifies various options. Continuing the example, an “accept” moves to a state where the desk clerk’s next question is “What type of room would you like?”; a “reject” stays in the same state, with the desk clerk saying that he didn’t understand; and a sequence of three rejects moves to a state where the clerk says he didn’t understand, but asks whether a room for one night will be okay. On reaching the end of the lesson, the student either exits or selects a new lesson from a menu.

The rest of the paper is organised as follows. Section 2 gives a brief summary of the technical aspects of the system, and Section 3 describes the actual course content. Section 4 discusses gamification, and Section 5 presents an initial pilot study. The final section concludes.

2 TECHNICAL BACKGROUND

In this section, we describe the main components of the system: grammar-based speech recognition and rule-based translation, lesson structure, user interface and web deployment.

2.1 Grammar-based speech and language processing

The speech and language processing components of the system are entirely rule-based, and have been developed using the Regulus toolkit [3]. The same grammar is used both to provide the language model which constrains the speech recogniser, and to parse the result of performing speech recognition. The intent is to ensure that all speech recognition results will be within the coverage of the language processing components, which greatly simplifies the architecture by obviating the need for robust parsing and semantic interpretation.

The Regulus toolkit sits on top of the commercial Nuance Recognizer platform, and provides a flexible mechanism for constructing efficient language models implemented in Grammar Specification Language (GSL), Nuance’s proprietary version of Context-Free Grammar (CFG). For each recognition language (in the case of the specific course used here, English), the Regulus platform provides a central resource grammar implemented using a feature-grammar formalism. The resource grammar is not compiled directly into a CFG language model. Instead, we first extract a domain-specific subgrammar using Explanation-Based Learning (EBL) methods driven by the set of example sentences supplied by the course designer; the scheme is explained in detail in [3], which also includes a thorough description of the English resource grammar.

The grammar, and hence the derived Nuance recognition package, produces semantic representations in Almost Flat Functional Semantics (AFF; [4]), a minimal formalism where clauses are represented as unordered lists of elements tagged with functional roles. For example, “Could you give me directions to the zoo?” is represented as the structure

```
[null=[utterance_type, ynq],
 agent=[pronoun, you],
 null=[modal, could],
 null=[action, give],
 null=[voice, active],
 object=[abstract, directions],
 indobj=[pronoun, i],
 to_loc=[loc, zoo]]
```

2.2 Using interlingua to display prompts

The AFF representations produced by speech recognition and parsing are translated into a language-neutral form, also expressed using AFF. The minimal list-based format means that translation rules can be very simple: basically, they map tagged lists onto tagged lists, possibly conditional on the surrounding context. The translation rule infrastructure is also part of the Regulus toolkit [4].

The space of permitted interlingual representation is defined using another Regulus grammar, which associates each valid interlingual AFF representation with a surface form. In the context of the CALL-SLT system, the intention is that these interlingual representations form a reduced, simplified and abstracted version of the English grammar, while the surface form is used as part of the prompt given to the student. Thus, continuing the previous example, the AFF form corresponding to “Could you give me directions to the zoo?” is converted into the interlingual representation

```
[null=[utterance_type, request]
 arg2=[abstract, directions],
 to=[loc, zoo]]
```

using a set of translation rules of which the least trivial is the one which maps the elements corresponding to “Could you give me...” to the single element `null=[utterance type, request]`.

We have experimented with several strategies for defining interlingua grammars: as usual, there are a number of competing requirements. On the one hand, we want the grammar to be structurally simple, so that the process of converting interlingual AFF representations to surface forms can be as fast as possible. A second requirement is that the surface forms should be fairly natural-looking L1 expressions. A third is that it should be easy to port the system to support a new L1.

The compromise we have found best in practice is to define a minimal interlingua grammar parameterized in two ways. First, there are hooks allowing generic changes to the default Subject-Verb-Object word order: for the specific case of German, relevant here, we want the modifiers to come before the verb if it is not the main verb (naturally, this is not the case in English). The grammar is kept very simple to make it feasible to allow this kind of flexibility: for example, we do not have any kind of moved question construction, so the word order is “You want what?” rather than “What do you want?”. The only concession to grammatical agreement is to add a formal affix to the verb, making it agree with the subject. The second type of parameterization is to handle surface forms. The plain grammar produces surface forms using English words; a final processing step uses a set of surface rewriting rules to map English words into the final L1.

Although the scheme lacks theoretical elegance, it is easy to implement and maintain, and performs well in practice. The examples below present some typical prompts taken from the English L2/German L1 version, showing the original English example, and the translated prompt both before and after surface rewriting.

> i would like a double room
> ask-for : double room
> Frag: Doppelzimmer

> can i pay by credit card
> say : i want-to 1-SING with credit-card pay INF
> Sag: Ich möchte mit Kreditkarte bezahlen

2.3 Providing help examples

When the student is uncertain how to answer a prompt, the system can provide them with an example of a correct response. The text forms of these examples come from the corpus of sentences supplied by the course designer, which is also used for producing the prompts. The associated audio files are collected by logging examples where users registered as native speakers got correct matches while using the system. Since every prompt is derived from an example taken from the L2 corpus, at least one example is always available. Given that the interlingua grammar used to define the prompts has less structure than the L2, and maps many L2 sentences onto the same prompt, there are typically several possible help examples available.

By recording successful interactions made by users registered as native speakers of the L2, the system can also store spoken help examples. When creating the course, a native speaker cycles through all the examples using a "recording mode" specifically designed to support this task, where the system only offers the user prompts for which recorded examples do not yet exist. Each speech example is tagged with the words found by the speech recogniser when it was recorded, so that students can be offered both text and speech help. Since slightly incorrect responses can still be counted as successful matches (most often, an article is inserted or deleted), a second pass is required to correct erroneous transcriptions. This is done efficiently by creating an editable HTML table, which includes both transcriptions and links to the associated speech files.

2.4 Lesson structure

For pedagogical reasons, it is desirable to group the examples into thematic units; we call these units "lessons". The most straightforward alternative, used in early versions of the system, did no more than divide up the corpus into a number of possibly overlapping sets, with each set corresponding to a lesson. The unifying theme of a lesson was either the language required to carry out a concrete task (booking a ticket, ordering at a restaurant, buying clothes), or one or more types of grammatical construction (numbers, dates, simple questions).

Grouping examples into lessons creates structure and makes the activity of practicing with the system feel more focussed and meaningful. This suggested to us that introducing further structure might be worthwhile. To this end, the version presented here associates a simple dialogue script with the lesson. The script, written in an XML notation, defines a number of steps, typically about 10 to 20; the specification of each step includes a unique ID, a group of prompts, a recorded multimedia file, and the steps to move to next depending on different conditions. For example, the XML code for the step in the Hotel script described above is as follows:

```
<!-- Ask for number of nights -->

<step>
<id>ask_for_number_nights</id>
<multimedia>how_many_nights</multimedia>
<group>room_for_number_of_nights</group>
<repeat>ask_for_number_nights</repeat>
<limit>is_one_night_okay</limit>
<success probability="25">not_available</success>
<success>ask_type_of_room</success>
</step>
```

We briefly explain the content. When the step is executed, the multimedia file `how_many_nights` shows a clip of a cartoon desk clerk asking the student how many nights they wish to stay for; the associated prompt will be taken from the group `room_for_number_of_nights`, and, rendered in the L1, will mean something like "request: room for (number) nights". The `repeat` tag says to repeat the step if the student's response is not accepted. If it is not accepted three times, the `limit` tag says to move to the step `is_one_night_okay`, where the student is asked a simple yes-no question. Conversely, if the response is accepted, the two `success` tags say to move either to the step `not_available` (25% probability) or otherwise to the step `ask_type_of_room`.

2.5 Web deployment and user interface

CALL-SLT is deployed over the web using a scalable architecture, developed by Paideia Inc, California and designed for cloud-based computing. In common with similar platforms, like WAMI [5] and Nuance's Mobile Developer Platform, it uses a client/server approach in which speech recognition is carried out on the server side. We incorporate architectures based on network speech recognition, as client/server approaches will still play an important role in the future for two main reasons: 1) speech recognition that is carried out on the server side can be combined with considerably more elaborate server-side functionality and 2) the cost of wireless networks will become increasingly lower in the near future.

The overhead due to web deployment is a few hundred milliseconds per recognition operation, compared to execution on a desktop machine. Full details are presented in [6]. By moving almost all processing to the server, the client can be kept simple enough that it can be comfortably hosted even on a smartphone. It only needs to be responsible for the graphical user interface, maintaining a small amount of state logic, performing recording and playback of audio, and requesting services from the remote peer. Versions of the client for standard browsers have been developed using Flash 11 in combination with ActionScript 3.0.

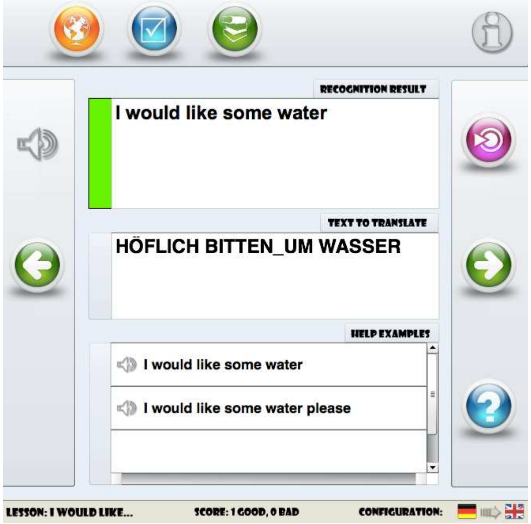
Due to the limitations of the target platform (lack of an endpointing mechanism), we have been forced to adopt a push-and-hold solution for initiating speech recognition, where users signify manually when they have stopped speaking to the system; specifically, users have to keep the mouse button pressed down while speaking. From our previous experiments, this has proven to be problematic. In order to attack this problem we introduced a more ergonomically comfortable approach for desktop browsers, where the press and release of a control keyboard button triggers each event.

3 COURSE CONTENT

Some novel aspects of the presented version of CALL-SLT are the inclusion of the teacher's opinion and feedback in the development process, the dialogue structure of the lessons, as described above, and the link to the textbook used in the school's curriculum. In the following sections we take a closer look at these components.

3.1 Teachers' contributions

In order to guarantee the development of a relevant CALL tool that can actually be of use for Swiss secondary schools, we gave the teachers the possibility of being involved from the very beginning and giving their inputs, both on the architecture and the content of the system. For the development of the version of the CALL-SLT system described here, we worked in close collaboration with the *Sekundarschule Gründen* in Muttenz, BL, Switzerland. We were in contact with the school's director and the school's English teachers for about a year, during which the tool was developed. At the beginning, the already existing version ([2]; cf. Fig. 1) of the system was demoed and in a subsequent discussion the teachers were asked how the tool could be adapted to their needs and how it could be made useful for their English classes. The inclusion of English teachers as subject matter experts is an extremely important point in our approach, because it is the only way to guarantee the satisfaction of the end-user.



I would like some water

HÖFLICH BITTEN_UM WASSER

I would like some water

I would like some water please

LESSON: I WOULD LIKE...

SCORE: 1 GOOD, 0 BAD

Figure 1: Interface of previous CALL-SLT version with plain text-based prompts

3.1.1 Textbook-based approach

In the teacher's opinion, one of the most important points was to create a link between the CALL-SLT tool and the school's English textbook and curriculum. The main textbooks used in the partner school are the “Ready For English” manual from the Klett publishing house [7] and the two workbooks “Speaking Tests, Englisch, Klasse 5-10” [8] and “Englisch London” [9]. During the discussions with the teachers it was decided to create eight lessons which loosely cover the first year’s class content and which could be put together to form a virtual trip to London in the end. The lessons cover the following topics: (1) Train station (course content = name, nationality, numbers, locations, time expressions), (2) Contact – Getting to know someone (course content = name, nationality, siblings, capitals), (3) Tube station (course content = numbers, locations, prices), (4) Hotel (course content = numbers, room types, prices, payment types, *where*-questions), (5) Tourist Information Office (course content = numbers, cultural knowledge of London, time expressions, ordinal numbers), (6) Restaurant (course content = food and beverages, payment types), (7) Asking and giving directions (course content = *where*-questions, directions, distances, cultural knowledge of London), (8) Shopping (course content = clothing, colours, numbers, like/dislike expressions). The specific content of these lessons was first put together in a corpus, which was then verified and adapted by the teachers. In a second step the corpus examples were connected into coherent lessons, which leads us to the second important input given by the teachers, discussed in the following section. The matching of the tool’s content with the textbooks means a major advantage of CALL-SLT compared to similar commercial tools, which follow their own logic and content structure and can therefore hardly be integrated into classroom teaching.

3.1.2 Dialogue-based approach

As mentioned before, another important change that was initiated by the teachers was to move from a disconnected text-based presentation to a coherent dialogue-based approach. For them it was essential that the students could engage in “real” conversations with English natives, since this allows them simultaneously to practise their receptive skills (by listening to the videos recorded by native English speakers) and their productive skills (by engaging in the conversation and responding to the questions). It has been emphasised in the “Common European Framework of Reference for Languages” [10] – *Interaction encompasses both receptive and productive activity as well as activity unique to the construction of joint discourse and therefore all reception and production strategies (...) are also involved in interaction* – and we also know from our own experience that these are two closely linked areas in SLA, and neither of the two components should be neglected in a CALL tool. Since the main focus of our approach lies on providing conversation scenarios – following the communicative approach to SLA – with which the language learners can train their communication skills, their fluency and pronunciation, as well as the learnt vocabulary and the utilisation of some acquired grammatical structures, the multimedia-prompting linked with the speech recognition technology seems to be the logical consequence.

In order to make the dialogue flow more interesting for the language learners, there are some variations in the dialogues, which can give the conversations an unexpected turn. It might be that a student is asked to order tickets for the musical “Mamma Mia” and then be told that this musical is already sold out, but that tickets for “Cats” could be offered. The dialogues are kept interesting by randomly choosing the variables for the responding prompts (in German), meaning that the same lesson can be practised several times, without having to repeat the same vocabulary. In future we probably want to add even more unexpected turns, to keep the dialogues interesting even after several runs through them.

After having developed a first prototype of the adapted system, the teachers again had the chance to test the tool in detail and give us feedback. Their inputs concerned a few functionalities in the architecture and some changes in the lesson design and coverage. Those changes were then integrated and a second testing phase began.

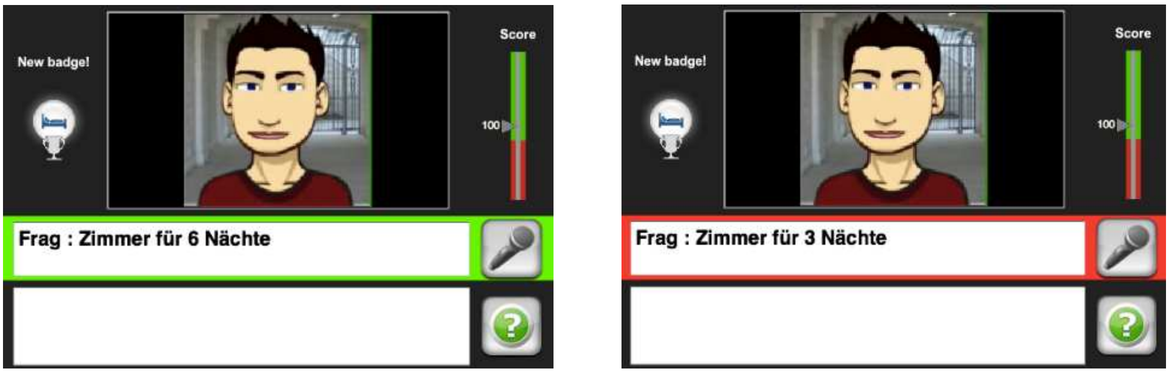
A major advantage is that, once a first set of lessons is created, the time and effort needed to create more lessons is reduced drastically, which also suggests that in future it will be possible for the teachers to design their own lessons, given that they have some basic IT knowledge and know how to write XML files. Failing that, it will also be easy for the developer to implement the teachers’ requests as we do now.

3.1.3 Pedagogical goals

From a pedagogical point of view, the CALL-SLT tool can fulfil various goals. On one side there are the purely linguistic aspects, meaning that the provided tool allows the pupils to effectively train their linguistic skills, such as vocabulary, grammar, pronunciation or fluency. On the other side we also want to provide a tool which allows the pupils to immerse into the cultural aspects that a language is always bound to. Both aspects were important to the collaborating teachers and are implemented in the current version of CALL-SLT.

3.2 How it works

The use of the tool is simple and intuitive. The students first log in with a personal username and password and can then choose the lesson with which they would like to train their English skills. Before using the tool for the first time it is advisable to watch the instructional video, which is also available on the tool's Internet page. In this video the students get short instructions on how to use the tool and on how the various functionalities work. When they are ready to start, they are shown a short video of a cartoon character, audio-recorded by a native English speaker, which represents the communication partner for the student. A typical lesson start would be "Hello". The student is always given a semantic value in his mother tongue (German in our case), which has to be included in the answer (e.g. Hallo). He then speaks his response (in English) into the microphone (e.g. "Hi"). While the speech recognition processes and evaluates the student's response, the spoken input is played back at the student, so that he can hear his own voice and notice potential mispronunciations. The system's feedback is given in form of a green (= successful recognition) or red (= misrecognition) frame around the variable box, as shown in Fig. 2.



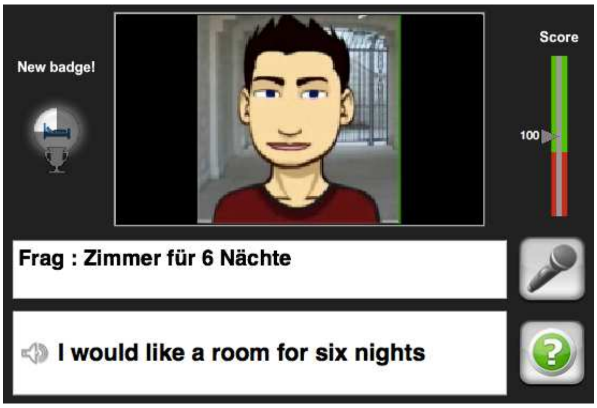
Frag : Zimmer für 6 Nächte

Frag : Zimmer für 3 Nächte

Figure 2: Recognition results showing a green frame for a successful recognition and a red frame for misrecognition.

3.2.1 Help function

If the student gets to a more advanced step and doesn't know how to answer correctly (e.g. Video = "How many nights would you like to stay at our hotel?"; Variable = "Frag: Zimmer für 6 Nächte"), he can use the help function (cf. Fig. 3), which suggests a straightforward example of a correct answer in both written and spoken form (recorded by a native English speaker), as for example "I would like a room for 6 nights".



Frag : Zimmer für 6 Nächte

I would like a room for six nights

Figure 3: The help function gives a written and spoken example of a correct response.

This function allows the student to get help if he doesn't know the prompted vocabulary or if he isn't sure how to pronounce the requested terms correctly. By imitating the help example, the chances of a successful recognition can be increased.

The help function is available throughout the lessons and always tries to supply one of the easiest versions of an answer for the example, so that the chances of succeeding increase and the student's motivation rises.

4 GAMIFICATION OF CALL-SLT

4.1 Motivation

In a pre-test conducted at a Geneva middle school one year ago, one of the major findings was that pupils of this age need additional motivation in order to play with the system in their leisure time. Due to this finding we introduced a scoring system and included some basic gamification elements, as they are described as a motivational instrument, e.g. in [11].

4.2 Gamification elements

4.2.1 Scoring system

The main element is the scoring mechanism itself. Pupils can gain points if they use more complex structures (e.g. "I want a Coke" = 1 point; "Could you please bring me a Coke" = 3 points) and they lose points if they use the help function or if they are misrecognized by the system. They start off with a credit of 100 points and they can always see their current score on a bar on the interface.

4.2.2 Badges

Another gamification element is the badge system as displayed in Fig. 4. The scoring system is directly linked to a badge system, in order to motivate the students to gain as many points as possible. They get a first, plain, badge for completing every lesson for the first time (not score-linked). For the bronze badge every lesson needs to be fully completed three times. The silver and gold badges are then linked to the scores. For the silver and gold badge the lessons need to be completed three times, with a minimal score of 85 points for silver and 90 for gold.

After every successfully completed lesson, the student has the choice between doing the same lesson again and collecting another badge or changing the lesson.

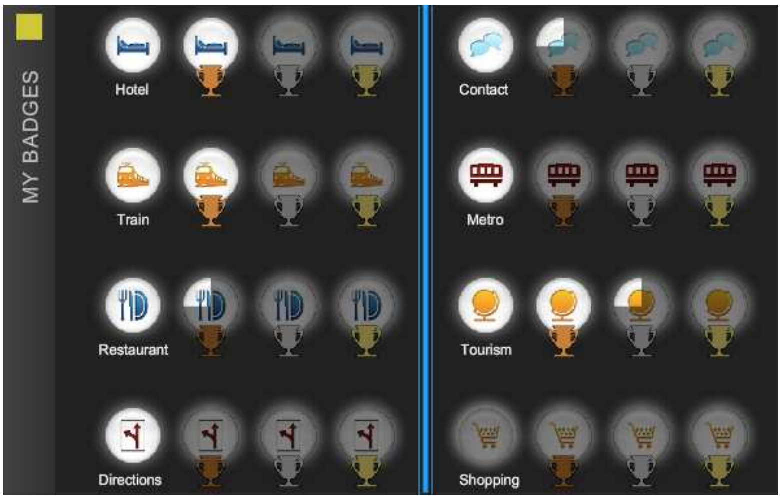


Figure 4: Badge system

5 EVALUATION/PILOT STUDY

We have used earlier versions of CALL-SLT to carry out several non-trivial evaluation exercises [12, 13, 14, 15, 16]. These provide good evidence that CALL-SLT can help students to improve their spoken language skills even over short time periods, and that system feedback coincides reasonably well with native speaker judgements. In particular, our most recent evaluation [17], carried out using a sample of 130 crowdsourced subjects recruited using the Amazon Mechanical Turk, shows that the results hold up in controlled experiments.

We have so far only had time to conduct an informal preliminary study on the German-English version described in the present paper; we asked several children aged between 9 and 13 to use the system for as long as they felt comfortable, while we observed them. We briefly note their reactions.

The younger children, who had no previous English, found the system challenging. Recognition did not work well for them, and they became somewhat frustrated. Except for a few very easy responses like “yes” and “hello”, they also used the help system throughout, imitating the spoken help examples. We were however pleased to see that they persevered for about an hour; towards the end, they were pronouncing noticeably better, which was reflected in the system accepting them more frequently. They often flipped to the “Badge” tab to see how many badges they had acquired, giving the impression that the badge system acted as an effective motivator.

The older children, who had already had a year or so of English at school, did much better. They found the system easy to use, only requiring a few minutes of instruction to become autonomous; after that, they were happy to use it for nearly two hours, doing one lesson after another in rapid succession. They seemed to find it enjoyable and even somewhat addictive. Speech recognition worked well for them, and after the first half-hour they often tried out spontaneous responses without first using the online help. Like the younger children, they appeared to be motivated by the badge system. On the negative side, their body language suggested that they found the system too slow and the multimedia files too long. There was also a tendency to use hyperarticulation after failed recognition, speaking in an unnaturally slow and distinct manner to make it easier for the system to understand. A related problem was that recognition appeared to be somewhat worse when they spoke in a polite way, e.g. adding “please” to the end of a request, which had the undesirable consequence of discouraging the use of politeness constructions.

We are feeding back experience from these initial trials into the next version of the system, which will be tested on a larger group of 12-year-old Germanophone school students in October and November 2013.

6 CONCLUSION AND FURTHER DIRECTIONS

In a future version of the CALL-SLT tool we will try to include some further elements to make the application more interesting and engaging for the users. One point, which has already been mentioned above, is the integration of more surprises in the dialogue design. Similar to video games, unexpected turns in the dialogues should help vary the experience in every run through the various lessons and maintain the learners’ motivation over a longer time period.

Another idea, which is linked to our gamification approach, is to include more game elements, such as a leaderboard, an avatar tutor or various difficulty levels. A leaderboard would increase the competitive character of the system and might at the same time increase students’ motivation to engage in this serious game. Even though leaderboards are a commonly used element in many computer and video games, we have to act with caution, considering that our target audience is teenage secondary school pupils. The least dangerous scenario would probably be to integrate an anonymous leaderboard, which simply shows how many badges have already been acquired for every lesson, or an indication of how many pupils are ahead and behind a specific player. This will help avoid a pedagogically counterproductive situation which might promote mobbing.

An avatar tutor might be included to provide help and/or tips and tricks to the students throughout the lessons. Such a tutor might give concrete feedback, if only a specific part of the student’s response is incorrect (e.g. “I would like a room for 5 nights” if the prompt said “Frag: Zimmer für 6 Nächte”), but also technical issues could be treated via an avatar tutor. If for example the microphone settings are incorrect, which prevents the response from being recorded, the tutor might tell the student to verify the microphone settings before continuing with the lesson.

Various difficulty levels could also help to make the tool yet more interesting and versatile. The various levels could for example be defined depending on the English accent used in the video clips. A standard British English might be used for the easy level and more challenging dialects, such as Scottish or Irish English could be used for more advanced levels.

7 ACKNOWLEDGEMENTS

The core CALL-SLT system has been developed under funding from the Swiss National Science Foundation and the Boninchi foundation. The web deployment architecture was developed by Paideia Inc, California. We would like to thank Nuance for allowing us to use their software for research purposes. Many thanks also to the collaborating teacher M. Pflugshaupt and Imyllia, Milla, Eline and Florian for acting as initial test subjects.

REFERENCES

[1] Wang, C. and S. Seneff. (2007) Automatic assessment of student translations for foreign language tutoring. In Proceedings of NAACL/HLT 2007, Rochester, NY.

[2] Rayner, M., P. Bouillon, N. Tsourakis, J. Gerlach, M. Georgescul, Y. Nakao, and C. Baur. (2010). A multilingual CALL game based on speech translation. In Proceedings of LREC 2010, Valletta, Malta.

[3] Rayner, M., B.A. Hockey, and P. Bouillon. (2006) Putting Linguistics into Speech Recognition: The Regulus Grammar Compiler. CSLI Press, Chicago.

[4] Rayner, M., P. Bouillon, B. Hockey, and Y. Nakao, (2008). "Almost flat functional semantics for speech translation". In Proceedings of COLING-2008, Manchester, England.

[5] Gruenstein, A., I. McGraw, and I. Badr. (2008). The WAMI toolkit for developing, deploying,and evaluating web-accessible multimodal interfaces. In Proceedings of the 10th international conference on Multimodal interfaces, pages 141-148. ACM.

[6] Fuchs, M., N. Tsourakis, and M. Rayner. (2012). A scalable architecture for web deployment of spoken dialogue systems. In Proceedings of LREC 2012, Istanbul, Turkey.

[7] Morrissey, F. A., H. Fäs, D. Marchini and D. Stotz (2006). Ready for English 1. Klett und Balmer AG, Zug, Switzerland.

[8] Ashworth, P., N. Sklenar, D. Vilimek, A. Weigl, J. Wood and I. Mayrhofer. (2009). Speaking Tests, Englisch, Klasse 5-10. Ernst Klett Verlag GmbH, Stuttgart.

[9] Petilliot-Becker, I. and R. Diehl. (2010). Englisch London, 7./8. Schuljahr. Cornelsen Verlag Scriptor GmbH & Co. KG, Berlin.

[10] Verhelst, N., P. Van Avermaet, S. Takala, N. Figueras, and B. North (2009). Common European Framework of Reference for Languages: learning, teaching, assessment.

[11] Werbach, K. and D. Hunter. (2012). For the Win: How Game Thinking Can Revolutionize Your Business. Wharton Digital Press

[12] Bouillon, P, I.S. Halimi, M. Rayner, and N. Tsourakis. (2011). Evaluating A Web-Based Spoken Language Translation Game For Learning Domain Language. in: Proceedings of the International Technology, Education and Development Conference, Valencia, Spain, 2011.

[13] Bouillon, P., M. Rayner, N. Tsourakis, and Q. Zhang. (2011). A student-centered evaluation of a web-based spoken translation game. In Proceedings of the SLaTE Workshop, Venice, Italy.

[14] Rayner, M., I. Frank, C. Chua, N. Tsourakis, and P. Bouillon. (2011). For a fistful of dollars: Using crowd-sourcing to evaluate a spoken language CALL application. In Proceedings of the SLaTE Workshop, Venice, Italy.

[15] Rayner, M., P. Bouillon, and J. Gerlach. (2012). Evaluating appropriateness of system responses in a spoken CALL game. In Proceedings of LREC 2012, Istanbul, Turkey.

[16] Jolidon, A. (2013). Reconnaissance vocale et amélioration de la prononciation : élaboration et évaluation de leçons avec le logiciel CALL-SLT. Masters Thesis, University of Geneva.

[17] Rayner, M. and Tsourakis, N. (2013). Methodological issues in evaluating a spoken CALL game: Can crowdsourcing help us perform controlled experiments? In Proceedings of the SLaTE Workshop, Grenoble, France.