

Content independent open-source language teaching framework

Randil Pushpananda

Language Tech. Research Laboratory
University of Colombo
School of Computing, Sri Lanka.
rpn@ucsc.cmb.ac.lk

Chamila Liyanage

Language Tech. Research Laboratory
University of Colombo
School of Computing, Sri Lanka.
cml@ucsc.cmb.ac.lk

Namal Udalamatta

Language Tech. Research Laboratory
University of Colombo
School of Computing, Sri Lanka.
ugn@ucsc.cmb.ac.lk

Ruvan Weerasinghe

Language Tech. Research Laboratory
University of Colombo
School of Computing, Sri Lanka.
arw@ucsc.cmb.ac.lk

Abstract

Language is a very powerful tool for mutual understanding between people. In the increasingly globalized society we live in, there is great importance and interest in learning languages other than one's own mother tongue. Computer based teaching tools provide a potent medium through which such demand can be met conveniently. *Shikshaka*, a computer based teaching framework was developed for teaching languages using a dialogue-based andragogy and is aimed to fulfill the above requirement. Effective technological methodologies were used to develop the system as an interactive tool using the goal oriented approach developed by linguistic scholars. We have used the framework to implement two scenarios: to teach Tamil using Sinhala and to teach Sinhala using English. The learner's language can be customized with little effort, while the framework is flexible enough to be customizable to teach other target languages as well with some pedagogical input.

1 Introduction

Listening, speaking, reading and writing are four main aspects of learning a new language. Due to the increased interest in learning new languages, language learners are demanding more and more user-friendly, less disruptive resources to fulfill this need. Globalization is the main reason for

the increasing interest in learning second, third and foreign languages. Ease of travel, advances in technology, and internationally focused economic systems are the benefits which can be obtained through language learning. In addition to the above, there are other motivating factors to learn new languages such as understanding alien cultures, increasing the number of people on the globe with whom you can communicate, and to make travel more feasible and enjoyable, among others (Rane and Sasikumar, 2007). The lack of knowledge in a second language on the other hand has been the cause of many misunderstandings and even causes for feuds and wars. Sri Lanka's ethnic conflict can be argued to have its roots in language among other factors. No local language project could ignore the strategic opportunity provided by technology to scale the teaching and learning of another language.

Learning a new language indeed is not an easy task for most humans. It requires proper guidance and materials. There are various kinds of materials available for teaching and learning languages, each using their own pedagogical and methodological approaches to teach language. Rane and Sasikumar (2007) showed that, learning new languages using books, magazines and courses conducted by the native language experts are traditional approaches, and that these traditional approaches do not meet some of the demands of modern life, which require *anytime* learning and *on-demand* learning.

Some of these requirements can be addressed effectively by developing computer based learn-

ing tools that can assist people in learning new languages. Effective use of such technology has the potential to overcome some of the limitations of traditional language learning contexts.

In Sri Lanka, effective use of computer technologies in the language learning process have not been reported in the literature. Primarily this is due to the fact that not much research work has been carried out in this area. Sri Lanka is a country with a multi-ethnic, multilingual and multi-script environment. In such an environment, learning several languages is advantageous and essential for the entire population. Considering these and other facts, we mainly focused our research on developing an open-source and customizable language teaching-learning framework to develop a Computer Assisted Language Learning (CALL) tool to teach any (target) language using any other (source) language. By using such a framework and as a proof of concept, we targeted to develop two language learning scenarios: to teach Tamil in Sinhala and Sinhala in English.

The rest of this paper is organized as follows. Section 2 summarizes the related work in this area; Section 3 describes the language teaching and learning concepts; Section 4 describes the methodology used to build the language teaching framework; Section 5 describes the design and implementation of the system; Section 6 describes the tools were developed using the teaching framework. Finally the paper concludes with a summary of the current status and looking future works.

2 Related Work

There is not much literature available on the use of CALL tools in teaching non-Latin languages. Most learning and teaching tools available are for teaching English as a second language using proprietary, closed systems.

Vsoft is an educational software development company in Sri Lanka which produces CALL software for the use of students who are preparing for various exams.

In addition to such CD-ROM based tools, there are some web based tools for teaching Sinhala such as *Let's Speak Sinhala*, *Learn Sinhala* (Nick and Nishanthi, 2008; *Vsoft*). These are targeted at helping adults and children (respectively) learn the basics of spoken Sinhala.

CALL tools for other languages such as Indic languages are available to some extent. *Marathi Tutor* is a web-based constructive learning envi-

ronment for teaching spoken or conversational Marathi. Rane and Sasikumar (2007) showed that the above framework covers basic vocabulary and construction of simple and compound sentences, thus enabling the learner to converse fairly well in common places such as at a bank or post office.

MarathiMitra (MarathiMitra) is also a commercial web based tutoring system for teaching spoken Marathi. In this system, the medium of instruction (source language) is English. This is neither an open-source nor a content independent framework.

RosettaStone (Rosetta) is also a commercial computer based language learning software, which supports some 20 languages using rich visual imagery to help students learn and think in a new language. Even though this tool supports Indic language such as Hindi, it still does not support Sinhala or Tamil.

The University of Bologna in Italy has developed a multimedia framework for second language teaching in self-access environments under the *DIAPASON* project. Tamburini (1999) showed that *DIAPASON* enables the teaching of English to university students up to an intermediate level, by building a self-access environment freely available to learners.

Many existing commercial and proprietary language learning tools do not support Unicode, are sometimes complex for non-technical learners to handle, and most importantly are not based on language teaching pedagogy. In addition, most of the above tools are hard-wired for teaching a particular language, and do not provide a framework for linguists to develop effective language learning environments.

3 Second Language Teaching and Learning

Second language teaching is a research area of Applied Linguistics. When someone learns a language as native in the speech community is called the language acquisition. Learn a new language other than the native we called second language or foreign language learning. These two terms can be misleading and it makes sense by their definitions. For example, Tamil and English languages are second languages for a native Sinhala speaker while Chinese and Japanese are foreign languages for them. Similarly languages like Sinhala and English are foreign languages for Japanese.

In applied linguistics, a broad range of research activities on second language teaching has been carried out. Pedagogical methodologies which have been identified in language teaching and learning were used in developing the *Shikshaka* framework. Contents developed for the framework are based on the study of second language learning pedagogy.

4 Methodology

4.1 Pedagogical consideration

The methodology we used to develop this framework can be divided into two steps. As the first step we studied the spoken language teaching methodology by examining the courses which were developed as textbooks (Gair et al., 2005; Karunatilaka, 2004; Fairbanks, 1968; Gunasekara, 2008) and computer based educational systems (RosettaStone, Nick and Nishanthi, 2008). It was clearly identified that the most effective way of teaching spoken language is through conversations at general places and situations such as at restaurants, post offices, police stations in order to facilitate the more effective *situated* kind of learning in adults. This situation based language learning methodology has been shown to lead to effective grasping of language constructs compared to formal abstract teaching. As the second step we studied how to increase the interactivity and attractiveness of the language tools and how to implement functionality for this within the framework. It was clearly identified that the use of real still images, callouts and audio files will increase the above two factors. Callouts were used to display the phonetic representation of utterances and are aimed at helping users with pronouncing words of the target language.

The framework has been designed in such a way that language teachers can add contents to the grammar section of the framework according to their preferences. In addition to the above, three types of exercises were introduced to practice the learning which gain through the lessons. It includes two word matching exercises and one sound matching exercise. Sound matching exercises are used to help learners identify words and pronunciation, while word matching exercises are used to help them identify different word classes.

4.2 Technical consideration

Our research identified that most of the frameworks and tools have some deficiencies such as

embedded content (i.e. content is hard-wired with the framework and they cannot be separated), non localizability, platform dependency, lack of Unicode support and difficulties of adding and modifying content.

Our research proposed a solution to address the above deficiencies and provide users with multimedia rich interactive language teaching environment. This is primarily achieved by separating the course content from the framework. XML technology was used to keep the course content independent so that the tool can be easily adapted and localized for the other languages and cultures. The framework also facilitates the changing of the sequence of lessons as required.

5 Design and Implementation

The content management framework and the user interfaces of the system are implemented using *Adobe Flash* technology. Flash provides state-of-the-art rich Internet application technology for developing interactive media rich tools over the web. *ActionScript* was used for communication between user interface and XML files in developing the on-demand loaded content for the flash application. The choice of Flash enables the delivery of an efficient, lightweight, platform independent, multimedia rich and web-enabled system. Proposed combined architecture which combines both Flash objects and XML files provides a flexible way of adding and deploying fully customizable content (lessons) rapidly. One important gain in using this combination of technologies was that it was able to overcome the issue of flash's lack of support for Sinhala Unicode rendering. This benefit accrue to any new language which involves complex scripts which are currently not supported by flash.

Since XML stores the information in a more accurate, flexible and adaptable way, the XML technology was used to hold data for the framework and maintain the structure of the data. The use of XML makes the content of the framework more readable and more extendable. Each type of course contents is organized in one or more XML files. The framework is engineered in a manner so that it can be readily customized for teaching other subjects in other languages.

Each of the lessons which include dialogues, the grammar and exercises can be separated into chapters from 1 to n . The *Shikshaka* framework was developed to support any number of sections (referred to as chapters) and is flexible enough to extend itself to be used by any two languages.

Each chapter (i.e. chapter folder) contains dialogue, grammar and exercise folders (Figure 1) and each folder type has its own file structure. Separate XML files are maintained inside each folder to reduce the complexity of the framework.

5.1 Dialogue

The framework is designed to use images, voices, text and callouts to make the dialogues more attractive (Figure 2). Users can customize these features according to their requirements.

Images: Since Human expressions can be shown clearly by using digital photographs we used digital images for conversations. The framework supports for still images with 1024x768 resolution and most of image file formats such as .JPG, .PNG and etc.

Voices: Exact pronunciations of words of both target language and source language are given by pre-recorded voice clips. Popular audio formats like .mp3, .wav and etc. can be used for voice clips. Name of the file, wave format and file path of voice clips can be changed by editing the XML file.

Text: Three types of texts are shown namely; text in target language script, text in source language script and the pronunciation of target language's words in transliterated form. All these features can be customized as user preferred.

Callouts: Callouts are used to show the text which mentioned above in an attractive way to the users. By editing the XML file, callout positions (x and y coordinates) and size of the callout can be changed.

XML file (dialogue): We designed an XML file for conversations (Figure 3) and maintained XML files for each and every conversation separately. It includes source language text, transliterated text, target language text, font sizes, font colors, font names, coordinates of the image file, size and name of the image, names of the source and target audio files, name of the callout and etc.

5.2 Grammar

The grammar being used in the dialogues can be elaborated in the grammar section by adding the relevant grammar in to the framework. The framework was designed to add titles and descriptions related to the title as two separate XML files and a separate cascading style sheet (CSS) was designed to handle the formatting of contents inside the XML data file.

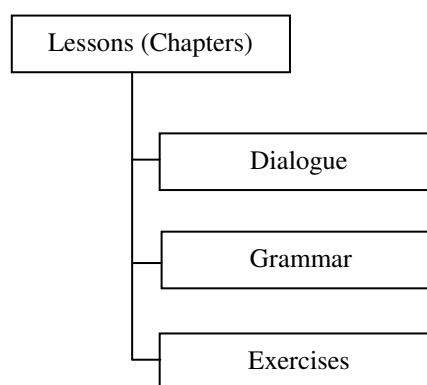


Figure 1. Main folder structure of the framework.

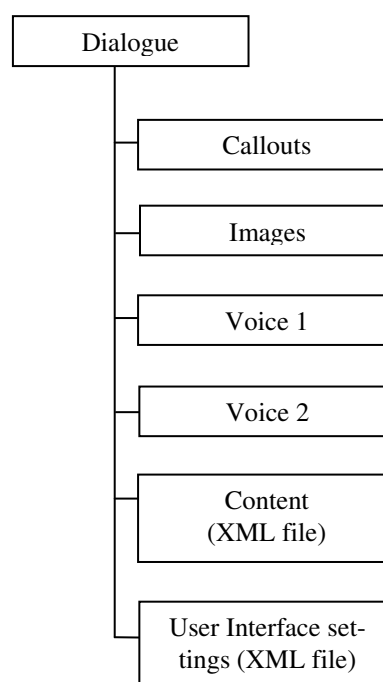


Figure 2. Structure of a dialogue.

```

<person name="பித்தர்">
  
  <audioclip lang="ta" src="vimal_1.mp3"/>
  <audioclip lang="si" src="vimal_s_1.mp3"/>

  <callout>
    
    <text lang="ta" x="585" y="228" height="100" width="200"
      fontname="Malithi Web" fontsize="35" fontcolor="0x0000FF">
      என்ன இருக்கிறது?</text>
    <text lang="tr" x="585" y="258" height="100" width="200"
      fontname="Malithi Web" fontsize="30" fontcolor="0x000000">
      එන්න ඉරික්කි ?</text>
    <text lang="si" x="585" y="288" height="100" width="200"
      fontname="Malithi Web" fontsize="30" fontcolor="0xFF0000">
      මොනවාද තිබෙන්නේ?</text>
  </callout>
</person>
  
```

Figure 3. XML structure for dialogue.

5.3 Exercises

The framework has designed to include relevant exercises of a particular chapter into the exercise section. Contents of the exercises are kept in separate XML files and three mapping exercises were designed as mentioned in the methodology section. To increase the attractiveness of the exercises, animated gif images were used to indicate correct and wrong answers.

6 Experiments

Two CALL scenarios were developed using the *Shikshaka* framework to teach Sinhala through English and Tamil through Sinhala. Figure 4 and Figure 5 show screenshots of the front page of the two case studies (refer Appendix A for examples of the user interface design of these courses). Developers or language teachers can use the *Shikshaka* framework to design the language learning tools conveniently for any target language using a source language.

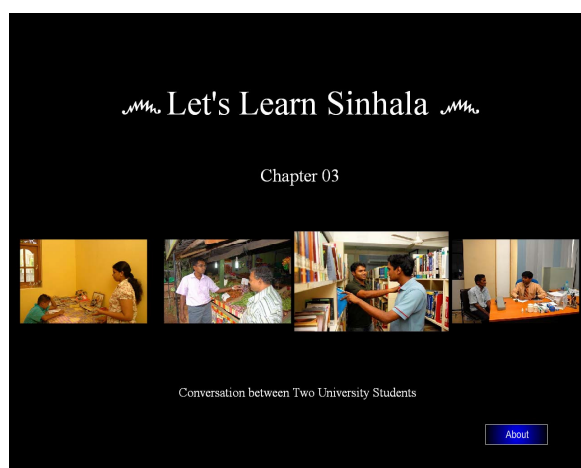


Figure 4. Sinhala learning tool.



Figure 5. Tamil learning tool.

These two scenarios were released under Creative Common public licensing (attribution, non-commercial and share alike) category. Samples of both tools were distributed freely in CD format especially in exhibitions and workshops to get feedback from users. School teachers have recommended distributing these tools through the school network as students can be motivated with IT enabled teaching methods. In addition to the above, both courses are publicly available at <http://www.e-learning.lk/vle/course/>. The courses are meant to be facilitated by a teacher, but may also be used for self-learning by the disciplined user. The full content together with the source code and documentation will also be available for download from the www.ucsc.lk/ltrl

7 Conclusions and Future Work

The *Shikshaka* framework was developed to address the problems identified in section 4.2 of most teaching tools being proprietary and hard-wiring content with framework. Two language learning courses were developed for teaching Tamil through Sinhala and Sinhala through English as a proof of concept for the framework.

Plans are underway to distribute the two courses in CD/DVD format in addition to their access through the web. Controlled offering of the courses are also planned in order to solicit feedback in order to evaluate the effectiveness of the framework and pedagogy. In addition to the above there is some demand to include an evaluation system to the framework for each lesson as well as revision exercises after each five chapters.

Moreover, it is expected to provide a more convenient content entry method for the framework without the need for editing coordinates and sizes of the XML data. This will help to improve the user-friendliness of the framework. Finally the *Shikshaka* framework will also be extended to support levels of study.

Acknowledgment

This work was carried out under PAN localization project phase 2 funded by IDRC Canada. We are immensely grateful to authors of the book; "An Introduction to Spoken Tamil" W. S. Karnatilaka, James W. Gair, and S. Suseendira-jah and the author of the translated version of the above book W. M. Wijeratne for providing us the contents for developing the Tamil learning tool. The contribution made by Tissa Jayawardena for reviewing the contents of the Sinhala learning

tool is also acknowledged. We would also like to thank our colleagues at the Language Technology Research Laboratory, UCSC who provided insight and expertise that greatly assisted this research, including those who have left us since then. We also thank all the staff at University of Colombo School of Computing (UCSC) for giving us great support to develop the two courses. Authors would also like to acknowledge the feedback given by two unknown reviewers to improve the quality of the work reported. Any remaining errors are our own.

References

- G. H. Fairbanks, J. W. Gair, M. W. S. De Silva. 1968. *COLLOQUIAL SINHALESE*. Cornell University.
- J. W. Gair, S. Suseendirajah, W. S. Karunatilaka. 2005. *An introduction to spoken Tamil*. Godage International publishers, Sri Lanka.
- A. M. Gunasekara. 2008. *A comprehensive grammar of the Sinhalese language*. Godage International publishers, Sri Lanka.
- W. S. Karunatilaka. 2004. *An introduction to spoken Sinhala*. M. D. Gunasena and Co. Ltd, Sri Lanka.
- MarathiMitra .
www.marathimitra.com
- Nick and Nishanthi. 2008. *Let's Speak Sinhala*.
<http://www.speaksinhala.com/>
- A. Rane and M. Sasikumar. 2007. *A Constructive Learning Framework for Language Tutoring*. In: Iskander, M. Innovations in e-learning, instruction technology, assessment, and engineering education. Dordrecht Netherlands: Springer.
- Rosetta.
www.rosettastone.com
- F. Tamburini. 1999. *A multimedia framework for second language teaching in self-access environment*. Computers & Education 32 (2): 137-149.
- Vsoft.
<http://vsoft.lk/>

Appendix A. Design of the tool

The Sinhala learning course consists of 15 chapters while Tamil learning tool consists of 25 chapters. Each chapter contains a dialogue, a grammar and an exercise. Figures 6, 7 and 8 show the designed user interface for dialogue, grammar and exercises respectively.



Figure 6. User interface – Dialogue

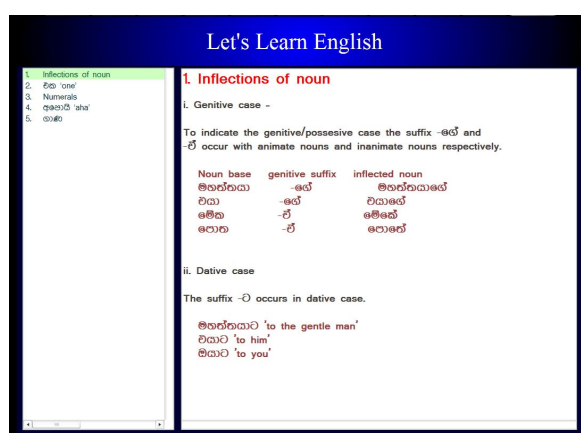


Figure 7. User interface – Grammar

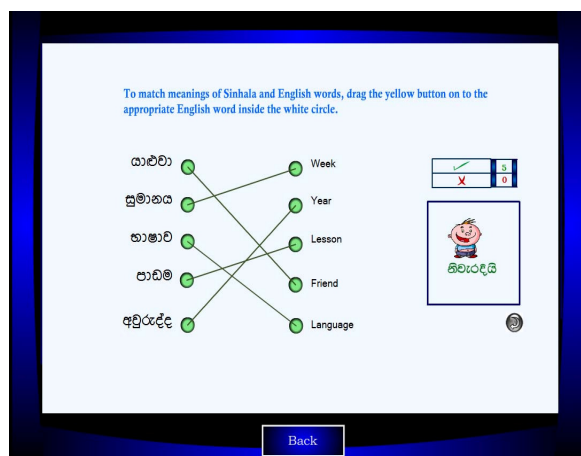


Figure 8. User interface - Exercise